Solving the Randomly Generated University Examination Timetabling Problem through Domain Transformation Approach (DTA)

Siti Khatijah Nor Abdul Rahim[1,2], Andrzej Bargiela[3], Rong Qu[4]

[1] Universiti Teknologi Mara Perak (Tapah Campus)
sitik781@perak.uitm.edu.my
[2] University of Nottingham (Malaysia Campus)
khyx8skn@nottingham.edu.my
[3] University of Nottingham (UK Campus)
abb@cs.nott.ac.uk
[4] University of Nottingham (UK Campus)
rxq@cs.nott.ac.uk

Abstract. Amongst the wide-ranging areas of the timetabling problems, educational timetabling was reported as one of the most studied and researched areas in the timetabling literature. In this paper, our focus is the university examination timetabling. Despite many approaches proposed in the timetabling literature, it has been observed that there is no single heuristic that is able to solve a broad spectrum of scheduling problems because of the incorporation of problem-specific features in the heuristics. This observation calls for more extensive research and study into how to generate good quality schedules consistently. In order to solve the university examination timetabling problem systematically and efficiently, in our previous work, we have proposed an approach that we called a Domain Transformation Approach (DTA) which is underpinned by the insights from Granular Computing concept. We have tested DTA on some benchmark examination timetabling datasets, and the results obtained were very encouraging. Motivated by the previous encouraging results obtained, in this paper we will be analyzing the proposed method in different aspects. The objectives of this study include (1) To test the generality/applicability/universality of the proposed method (2) To compare and analyze the quality of the schedules generated by utilizing Hill Climbing (HC) optimization versus Genetic Algorithm (GA) optimization on a randomly generated benchmark. Based on the results obtained in this study, it was shown that our proposed DTA method has produced very encouraging results on randomly generated problems. Having said this, it was also shown that our proposed DTA method is very universal and applicable to different sets of examination timetabling problems.

Keywords: Examination Scheduling, Domain Transformation Approach, Granular Computing, Randomly Generated Problem.

1 Introduction

Timetabling can be defined as a process of generating timetables or schedules that contain information about some events and the times at which they are planned to take place. Timetabling is normally considered a tiresome and laborious task. In some organizations, the personnel responsible in preparing the timetables usually do it manually and mostly using a trial-and-error approach.

Amongst the wide-ranging areas of the timetabling problems, educational timetabling was reported as one of the most studied and researched areas in the timetabling literature. Example of educational timetabling includes school timetabling (course/teacher timetabling), course timetabling, examination timetabling and etc.

In this paper, our focus is the university examination timetabling. Many universities in the world nowadays offer modular courses (across faculties) to their students, resulting in very strong interdependencies between students and exams data (many-many relationships in the data files). This has contributed to the complexities of the problem, and there were many claims in the literature that university timetabling problem as an NP complete problem [6], [7]. Besides satisfying hard constraints which is necessary to make the timetables feasible, the general objective of examination timetabling that is widely used in the literature is to reduce the cumulative inconvenience implied by the temporal proximity of consecutive exams taken by individual students. Even with the extensive research done in this area, there is still room for improvements for the state of the art methods especially in making sure the methods always able to reproduce good quality timetables consistently.

In the university examination timetabling research, the quality of the timetable is normally referred to as a‘cost’ which is measured using objective function proposed by Carter (Carter et. al, 1996) as below:

$$\frac{1}{T}\sum_{i=1}^{N}\sum_{j=i+1}^{N} s_{ij}\, w_{|\mathrm{pj}\ -\ \mathrm{pi}|} \qquad (1)$$

where N is the number of exams, Sij is the number of students enrolled in both exam i and j, pj is the time slot where exam j is scheduled, pi is the time slot where exam i is scheduled and T is the total number of students. According to this cost function, a student taking two exams that are | pj - pi | slots apart, where | pj - pi | ={1, 2, 3, 4, 5 }, leads to a cost of 16, 8, 4, 2, and 1, respectively.

2 Review of the Proposed Approach

In order to solve the university examination timetabling problem systematically and efficiently, we have proposed an approach that we called a *Domain Transformation Approach (DTA)* which was described in detail in our previous publications [10], [13]. DTA is an approach which is underpinned by the insights from Granular Computing concept [1], [2], [3], [4], [14]. In the proposed DTA, the examination scheduling problem is transformed into smaller sub problems, therefore is easier to be solved systematically and efficiently.

Our previous works [9], [10], [11], [12], [13] have tested DTA on some benchmark examination timetabling datasets, and the results obtained were very encouraging. We have managed to produce good and feasible quality timetables on all datasets in the experiments. Besides, our proposed DTA which avoided the utilization of random-search has resulted in the generation of consistent timetables (able to be reproducible) and shown a very deterministic optimization pattern.

3 Objectives of the Study

Motivated by the encouraging results obtained as stated in the previous section, we will be analyzing the proposed method in different aspects. In this section, we will list the objectives of doing the study in this paper. Each objective is elaborated in brief as below.

a) To test the generality/applicability/universality of the proposed method by testing it on a randomly generated benchmark datasets.
   Note: In the previous works, the proposed DTA was tested on benchmark datasets that were extensively tested by other researchers. However, it is agreed with [8], that when certain benchmark datasets are relied upon to evaluate an algorithm, the resulting algorithm could be inclined towards the criteria of the benchmark datasets. Therefore by testing on a randomly generated datasets, the universality of the method could be analyzed.

b) To compare and analyze the quality of the schedules generated by utilizing Hill Climbing (HC) optimization versus Genetic Algorithm (GA) optimization on a randomly generated benchmark datasets (with the hypothesis that HC will maintain to outperform GA based on previous outcome).
   Note: Minimization of the schedule cost by slot swapping is a Hill Climbing (HC) procedure which was implemented in the proposed DTA. Good quality feasible examination schedules were generated successfully using this method. In order to analyze whether a global search procedure could improve the quality of the schedules generated, we have also implemented Genetic Algorithm search procedure and incorporated it into the proposed DTA framework. It was observed that for Toronto benchmark datasets, HC has outperformed GA on all instances [11], [13]. Hence, we will see whether the good performance of HC is consistent on the newly tested datasets.

4 Domain Transformation Approach (DTA)

In our proposed DTA, after retrieval of datasets, data verification and standardization is done to alleviate the initial problem of dataset and format variety and convert it into a standard format that will be used as an input to the pre-processing stage. The pre-processing of data and constraints from the original problem space will provides important information granules which consecutively provide valuable information for scheduling. The new aggregated data (which was explained in greater detail in [10], [11], [12]) will reduce the subsequent cross-checking and cross-referencing in the original data thus expediting scheduling stage.

In the scheduling stage, which utilizes an allocation method using a Graph Colouring Heuristic coupled with a backtracking procedure (a modified version of Carter et al. (1996)'s backtracking approach [5]), is adopted as a basic scheduling process ([10], [11], [12]). It is expected to produce only feasible solutions with a total number of slots that will satisfy the minimum requirements given in the problem. The last stage in the proposed algorithm is the optimization stage. This stage involves three procedures: minimization of the overall slots conflict, minimization of schedule cost by slot swapping and minimization of schedule cost by reassigning exams.

Our previous proposed examination timetabling method has generated very encouraging results (i.e. good quality timetables) on all examination timetabling benchmark datasets tested [9], [10], [11], [12], [13]. However, in this paper our aim is to further analyze the proposed method as stated in the objectives section and for this purpose we will be focusing on the procedure to minimize the schedule cost by slots swapping. This is because it was observed that out of the three optimization procedures, the slot swapping has recorded the most significant reduction of the cost.

4.1 Minimization of the Schedule Cost by Slot Swapping

Optimizing the feasible examination schedule obtained by the scheduling stage by doing slots swapping is explicitly focused on minimization of the cost function. This process is also known as permutations of exams slots. Below, with the aid of some diagrams, we will demonstrate and briefly explain how slot swapping can reduce the schedule cost and hence increase the quality of the schedule.

As can be seen in Fig. 1, there are a few students: S1, S2, S3, S4 ,S5 …. Sn and a few exams: E1, E2, E3, E4 …Em together with a few time slots: T1, T2, T3, T4 …. Tk. In this example, student S1 has registered for exams E1, E2 and E4; and student S2 has registered for E3 and E4. Therefore, exams E1, E2 and E4 are the set of conflicting exams for student S1 and because of this, they cannot be assigned to the same time slots. The diagram below shows that these three exams are not assigned to the same time slot (they are assigned to time slots T1, T2 and T3 respectively) and thus this is considered a feasible examination schedule.

According to the above example, although student S2 has a feasible examination schedule, the timetable does not satisfy the soft constraint in terms of putting a gap between one exam and the next exam that student will have to sit. This is not necessary but satisfying this would improve the schedule quality by benefiting the student, since it allows the student to have more revision time between exams. Thus, if exam E3 is now reassigned to time slot T1, the quality of the schedule can be improved, as illustrated in Fig. 2.

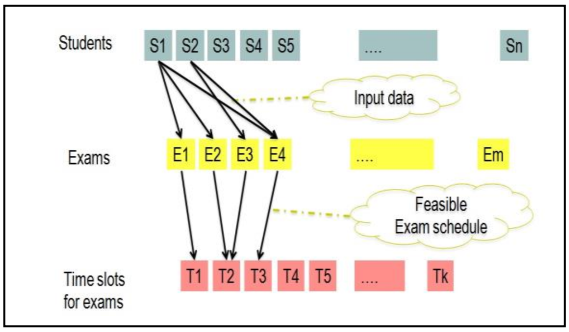

Fig. 1. An Example of a Feasible Examination Schedule.

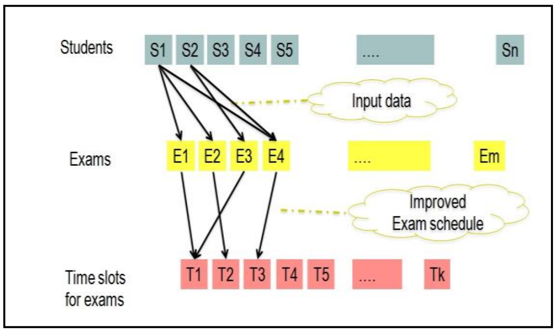

Fig. 2. An Example of an Improved Examination Schedule.

Fig. 3 below shows how the permutation of exam slots has changed the original ordering of the slots in Fig. 2, and consequently an improved schedule has been generated. By this permutation, a time slot has been added between time slot T2 and T3, and thus giving extra time for the students to do their revision.

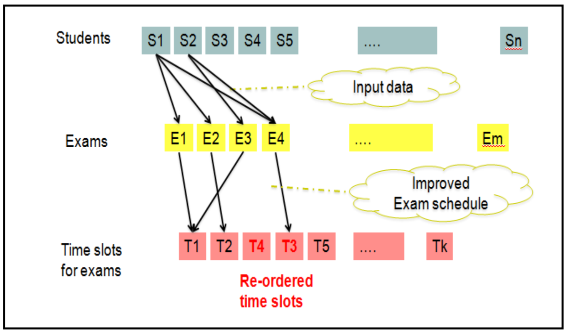

Fig. 3. An Improved Examination Schedule after Optimization (Permutations of Exam Slots).

However, adding an extra time slot between T1 and T2 will have a greater effect as illustrated in Fig. 4 than adding it between time slot T2 and T3 as illustrated in the previous diagram. In this new example, both students have more time between exams as compared to the previous example, hence the newly generated timetable is considered a better quality timetable.

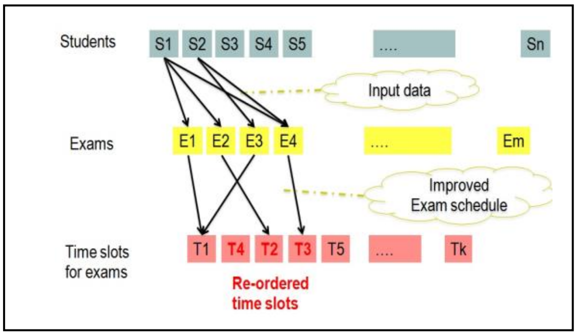

Fig. 4. Re-ordered Time Slots Via Permutations of Slots with Greater Effect

Following are the brief descriptions of the two types of permutations of slots.

Permutations of Slots: Hill Climbing Optimization. Shuffling the exam slots has the potential to reduce the cost of the schedule. This can be done by doing permutations of exam slots in the spread matrix [9], [10], [11], [12], [13]. Spread matrix will provide the information on how many students taking an exam from slot 'i' and 'j'. Permutations process involves the shuffling of slots or columns as block shifting and swapping. Each slot will be swapped with another slot in the provisional swapping stage, where by the Carter cost (1) is evaluated. The swap will be remembered and the exam proximity matrix will be updated if the swap resulted in a cost reduction. This kind of optimization procedure is called a greedy Hill Climbing (HC) because if a swap operation manages to improve the cost function, the swap is straight away accepted and the exam slots were rearranged accordingly. A few repetitions of block shift and swapping are done, besides restarting the optimization from several initial orderings of exam slots, in order to ensure that the greedy optimization does not lead to local optima.

Permutations of Slots: Genetic Algorithm Optimization. Realizing that the above proposed Hill Climbing which is a kind of a local search procedure may not direct the search to global optima, we have also implemented Genetic Algorithm (GA) to be integrated (substitute HC) into the proposed DTA. Having said that, this indirectly indicates that DTA is a flexible framework where different kinds of search can be used in one of the stage while maintaining other procedures (before and after the search) in timetable generation. Though GA is considered an 'old-fashioned' optimization procedure, but it has been confirmed that hybridizations of GA with some local search have led to some success in this area [15].

5 Experimentations, Results and Conclusion

We have tested the proposed DTA on the randomly generated university examination timetabling problem which was obtained from the University of Nottingham website (http://www.cs.nott.ac.uk/~rxq/data.htm). The following table (Table 1) listed the characteristics of each problem.

Table 1. Characteristics of the Randomly Generated Problems (Small Problems).

(a) Name of Dataset; (b) No of Exams; (c) No of Students; (d) No of Enrollments; (e) Conflict Density (f) Required No of Slots;

(a)	(b)	(c)	(d)	(e)	(f)
SP5	80	66	194	7%	15
SP10	100	100	359	11%	15
SP15	80	81	314	17%	15
SP20	80	83	344	19%	15
SP25	80	119	503	26%	15
SP30	80	126	577	32%	15
SP35	100	145	811	36%	19
SP40	81	168	798	42%	19
SP45	80	180	901	47%	19

The datasets were used in our solution as an input and we have obtained the following results as summarized in Table 2. We have observed that HC has outperformed GA in all problems as predicted according to the results obtained in the previous study [11], [13]. Even though HC is a local search procedure compared to GA, the repetitions of block shift and swapping, and restarting the optimization from several initial orderings of exam slots, have managed to ensure that the greedy optimization does not end in local optima.

Based on the results obtained in this study, it was also shown that our proposed DTA method has produced very encouraging results on randomly generated problems. Having said this, it was also shown that our proposed DTA method is very universal and applicable to different sets of examination timetabling problems. Besides generating good quality timetables, the DTA always produced consistent performance and demonstrate deterministic optimization pattern on all problems.

Table 2. Results For Randomly Generated University Examination Timetabling Problems Using Hill Climbing (HC) and Genetic Algorithm (GA) in the Optimization Stage of DTA

Problems	**SP5**	**SP10**	**SP15**	**SP20**	**SP25**	**SP30**	**SP35**	**SP40**	**SP45**
HC	3.4242	10.8900	16.0617	18.9277	23.5042	32.4762	45.2345	27.2083	29.9778
GA	4.1212	12.0000	16.8025	20.0723	25.7899	33.4365	47.6552	28.5714	33.1889

References

1. Bargiela, Andrzej, and Witold Pedrycz. Granular computing: an introduction. Springer Science & Business Media (2003)
2. Bargiela, Andrzej, Witold Pedrycz, and Kaoru Hirota. Logic-based granular prototyping. Computer Software and Applications Conference, 2002. COMPSAC 2002. Proceedings. 26th Annual International. IEEE (2002)
3. Bargiela, Andrzej, Witold Pedrycz, and Kaoru Hirota. Granular prototyping in fuzzy clustering. Fuzzy Systems, IEEE Transactions on 12.5 (2004): 697-709 (2004)
4. Bargiela, Andrzej, and Witold Pedrycz. Toward a theory of granular computing for human-centered information processing. Fuzzy Systems, IEEE Transactions on 16.2 (2008): 320-330 (2008)
5. Carter, Michael W., Gilbert Laporte, and Sau Yan Lee. Examination time-tabling: Algorithmic strategies and applications. Journal of the Operational Research Society (1996): 373-383 (1996)
6. Cooper, Tim B., and Jeffrey H. Kingston. The complexity of timetable construction problems. Springer Berlin Heidelberg (1996)
7. Even, Shimon, Alon Itai, and Adi Shamir. On the complexity of time table and multi-commodity flow problems. Foundations of Computer Science, 1975., 16th Annual Symposium on. IEEE (1975)
8. Lewis, R. A survey of metaheuristic-based techniques for university timetabling problems. OR spectrum, 30(1), 167-190 (2008)
9. Rahim, Siti Khatijah Nor Abdul, Andrzej Bargiela, and Rong Qu. Granular Modelling Of Exam To Slot Allocation. ECMS (2009)
10. Rahim, Siti Khatijah Nor Abdul, Andrzej Bargiela, and Rong Qu. Domain transformation approach to deterministic optimization of examination timetables. Artificial Intelligence Research 2.1 (2012): p122 (2012)
11. Rahim, Siti Khatijah Nor Abdul, Andrzej Bargiela, and Rong Qu. Hill climbing versus genetic algorithm optimization in solving the examination timetabling problem. 2nd International. Conference On Operations Research and Enterprise Systems, ICORES 2013 (2013)
12. Rahim, Siti Khatijah Nor Abdul, Andrzej Bargiela, and Rong Qu. Analysis Of Backtracking In University Examination Scheduling. ECMS (2013)
13. Rahim, Siti Khatijah Nor Abdul, Andrzej Bargiela, and Rong Qu. A Study on the Effectiveness of Genetic Algorithm and Identifying the Best Parameters Range for Slots Swapping in the Examination Scheduling. iSMSC (2013)
14. Pedrycz, Witold, Michael H. Smith, and Andre Bargiela. A granular signature of data. Fuzzy Information Processing Society, 2000. NAFIPS. 19th International Conference of the North American. IEEE (2000)
15. Qu, Rong, et al. A survey of search methodologies and automated system development for examination timetabling. Journal of scheduling 12.1 (2009): 55-89 (2009)